AN ABSTRACT OF THE THESIS OF

Ahmad M. Osta for the degree of Master of Science in Electrical and Computer Engineering presented on April 17, 1990.

Title : Overcoming the Singularity Problem in Digital Image Restoration.

Redacted for Privacy

Abstract approved by : _____
Prof. Roy Rathja

A recorded image is a degraded version of an original image. The process of removing the degradations is called restoration. Two types of degradations are considered in this thesis: namely, linear motion and defocused lens blurrings. Using the direct deconvolution technique to restore blurred images is impossible if the blurring matrix does not have an inverse. Such matrix is called a singular matrix. Facing this problem, researchers dropped the deconvolution approach and resorted to other algorithms such as the pseudo-inverse, a neural network, and a regularized iterative approaches.

It is the aim of this work to propose a new heuristic algorithm that overcomes the singularity problem using direct deconvolution and an iterative scheme. To test my algorithm against the other three algorithms computer simulations were performed, on blurred images, in a noise-free and noisy environments. It was found that in a noise-free environment my algorithm is superior to the other algorithms in terms of error and time consumption. In a noisy environment, my algorithm fails to restore comprehensible images.

Overcoming the Singularity Problem in Digital Image Restoration

by

Ahmad M. Osta

A Thesis
submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Completed April 17, 1990

Commencement June 1990

APPROVED :

Redacted for Privacy

Associate Professor, Electrical and Computer Engineering in charge of major

Redacted for Privacy

Head of Department of Electrical and Computer Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis is presented April 17, 1990.

Typed by Ahmad Osta for Ahmad M. Osta.

TABLE OF CONTENTS

1. INTRODUCTION 1
DEGRADATION TYPES 2
A PRIORI AND A POSTERIORI KNOWLEDE 3
THE SINGULARITY PROBLEM 4
CONSIDERATIONS AND GOALS 5
OVERVIEW OF THE FOLLOWING CHAPTERS 5

2. BACKGROUND 7
FORMULATION OF THE RESTORATION PROBLEM 7
EXPLORING IMPORTANT FEATURES OF SIPSF [H] 9
PROBLEMS OF RESTORATION 12
APPROACHES FOR RESTORATION 13
PSEUDO-INVERSE FILTER 14
NEURAL NETWORK APPROACH 15
REGULARIZED ITERATIVE APPROACH 18

3. PROPOSED ALGORITHM AND SIMULATIONS 22
PROBLEM STATEMENT 22
SINGULARITIES OF UNIFORM BLUR MATRICES 23
MY APPROACH TO THE SOLUTION 27
STEPS OF THE ALGOTITHM 28
FLOW CHART OF MY ALGORITHM 29
COMPUTER SIMULATIONS 31
RESTORING NOISY IMAGES 35
QUANTITATIVE RESULTS 35

4. DISCUSSION 41
PERFORMANCE CRITERIA 41
QUANTITATIVE RESULTS OF SIMULATIONS 42
COMPARISONS 44
MY ALGORITHM 44
DISADVANTAGES OF OTHER ALGORITHMS 46
THE EFFECT OF NOISE 47

5. CONCLUSION 48

BIBLIOGRAPHY 50

LIST OF FIGURES

Figures		Page
1.	The PSF of linear motion blur in the frequency domain	24
2.	The PSF of defocused lens blur in the 3-dimensional frequency domain	26
3.	Flow chart of my algorithm	30
4.	a) upper : the original image "ahmad"	32
	b) lower left : the linear motion blurred image	32
	c) lower right : the defocused lens blurred image	32
5.	Linear motion case, the restored images:	
	a) upper left : using pseudo-inverse filter	36
	b) upper right : using neural network approach	36
	c) lower left : using regularized iterative approach	36
	d) lower right : using my algorithm	36
6.	Defocused lens case, the restored images:	
	a) upper left : using pseudo-inverse filter	37
	b) upper right : using neural network approach	37
	c) lower left : using regularized iterative approach	37
	d) lower right : using my algorithm	37
7.	The restored images using my algorithm:	
	a) upper left : after 2 iterations	38
	b) upper right : after 3 iterations	38
	c) lower left : after 4 iterations	38
	d) lower right : after 5 iterations	38
	e) upper left : after 6 iterations	39
	f) upper right : after 7 iterations	39
	g) lower : after 8 iterations	39
8.	Noisy images	
	a) upper: noisy defocused lens blurred image (SNR=50 dB)	40
	b) lower left : restored image using neural network approach	40

LIST OF FIGURES (continued)

Figures		Page
8.	c) lower right : restored image using regularized iterative approach	40
9.	The convergence of my algorithm	46

LIST OF TABLES

Tables		Page
1.	The quantitative performance in case of linear motion deblurring using different algorithms.	42
2.	The quantitative performance in case of defocused lens deblurring using different algorithms.	43
3.	The performance of my algorithm after each iteration	43

OVERCOMING THE SINGULARITY PROBLEM IN DIGITAL IMAGE RESTORATION

1. INTRODUCTION

Images are produced to provide useful information about a phenomenon of interest. Unfortunately, since physical imaging systems are not perfect, a recorded image is a degraded version of the original image. In order to make use of the recorded image, it needs to be freed from its degradations. The process of removing the degradations is called restoration.

The concepts of restoration are relatively recent in origin due to the need for usually large-scale computing facilities. Restoration, nowadays, is widely used in many areas such as medical imagery, military technology, space imagery, and communications [1].

Sometimes the restoration concept is confused with the enhancement concept while both are different. Restoration is the process of removing degradations and restoring the original image. Enhancement is the process of placing the recorded image in a form suitable for our purposes even if it is going to emphasize one part of the image over another [2].

Although the image itself is a continuous phenomenon, the measuring, recording and manipulating instruments are all discrete. By discretizing an image, a cost-effective high technology, such as computers, could be used to process it as well as well-developed

mathematical tools such as linear algebra. When the discrete values of an image are quantized to specific values in a specific range, they are called pixels. Thus, a digital image could be represented in a matrix form with each element representing a pixel.

DEGRADATION TYPES

There are two main types of degradations, namely, deterministic and statistical. Statistical degradaton is due to noise. The source of such noise could be the imagery system, the medium, the recording process itself or a combination of all of these. Due to the random nature of noise, only some of its features, such as the first and second-order moments, can be calculated or measured.

On the other hand, deterministic degradation is a deterministic process that could be analyzed and quantitized. Such degradation could be caused by problems such as linear motion, atmospheric turbulence, defocused lens, or optical system aberrations. This type of degradation is called blurring. The mathematical function that represents a blur is called the point-spread function (PSF) of the image formation system [1].

It is possible to have a PSF that varies across the image; hence the system is said to possess a space-variant point-spread function (SVPSF). On the other hand, an image formation system may act uniformly across the image, i.e independent of position, and hence it has a space-invariant point-spread function (SIPSF) . While the latter is amenable to simple mathematical manipulation, as will be

seen later, the former is more complex in terms of mathematical manipulation, time consumption and restoration algorithms.

For this thesis, only blurs of the SIPSF type will be considered. Moreover, the degradations that are to be used are linear motion blur and defocused lens blur.

A PRIORI AND A POSTERIORI KNOWLEDGE

In restoration, some type of information about original images and imagery systems should be available. Such information could be the first and second-order moments (mean value, auto correlation or power spectral density) of the original image, or a general description of it. The noise variance of the imagery system is other information that could be needed. On the other hand, the availability of the PSF of the imagery system is a necessity.

When the needed information is available before processing any image, then it is called *a priori* knowledge. This kind of knowledge is possible when the imagery system and the corresponding environment are known. This availability allows us to study and analyze the system either through test pattern usage or mathematical analysis. For instance, the PSF of a camera-film system could be measured through measuring the PSF of its lens and film.

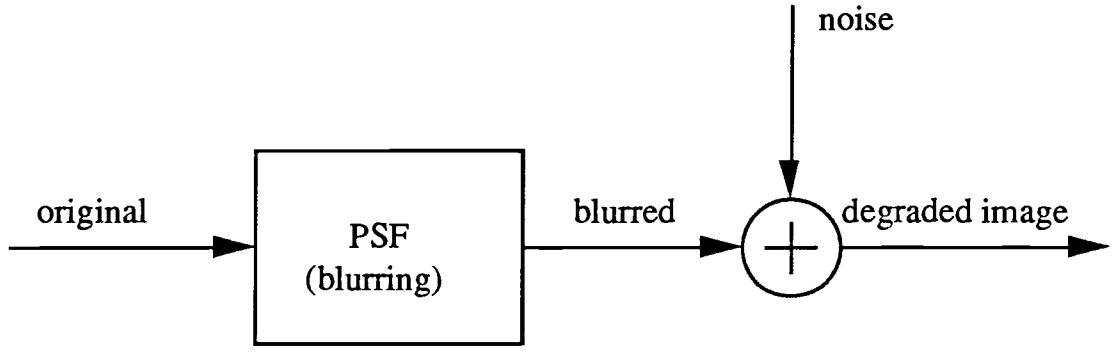
On the other hand, not having *a priori* knowledge for any reason would require working out the needed information from the blurred image. This would provide *a posteriori* knowledge.

Techniques for measuring PSF's from blurred images are documented in the literature [1,2]. Usually, edges and light point sources such as stars are used for PSF estimation.

The knowledge at hand, whether it is *a priori* or *a posteriori*, would be used in restoration algorithms as constraints on the solution or as criteria which the solution should satisfy.

THE SINGULARITY PROBLEM

The blurring process could be modeled as a convolution between the PSF of the imagery system and the original image [1]. If noise is involved in the recording process, noise can be represented as an additive uncorrelated term in the model as shown below



In a noise-free environment, the degradation process is just a convolution process. Thus, knowing the PSF of the imagery system, deconvolution is simply the answer for restoration. Unfortunately, this is not true all the time. The deconvolution solution utilizes the inverse of the PSF matrix, and when the inverse does not exist, deconvolution cannot be used. When the inverse of a PSF matrix

does not exist, then it is called a singular matrix. The singularity problem is common in digital image restoration [1].

CONSIDERATIONS AND GOALS

For this thesis, noise degradation is not considered. In other words, the environment is noise free. Moreover, only two kinds of blur degradations are considered: namely, linear motion and defocused lens degradations.

Under these kinds of degradations a perfect restoration could be achieved if the singularity problem does not exist. To go around the singularity and other problems, some researchers adopted approximate solutions with high time consumption algorithms.

It is the aim of this work to present an algorithm that overcomes the singularity problem. This proposed algorithm would restore, perfectly, images that are blurred by linear motion or defocused lens, without approximations and in less time.

OVERVIEW OF THE FOLLOWING CHAPTERS

The thesis is divided into five chapters. The following is a brief description of their contents.

In chapter 2, the restoration problem is formulated. Problems facing researchers are stated. Three current algorithms, to be used later for performance comparison, are briefly formulated and listed.

In chapter 3, the singularity problem in blur matrices is investigated. My proposed algorithm is examined. Simulations for my algorithm and three other algorithms are performed.

In chapter 4, the performance criteria and equations are discussed. The results of simulations are tabulated and discussed. The advantages and disadvantages of each used algorithm are presented. Also, the effect of noise on my algorithm is pointed out.

Chapter 5 concludes the research by summarizing the achievements and their importance as suggested by the performance comparisons. Also, it indicates the potential of my algorithm to be extended for restoring noisy blurred images. This is left for future research.

2. BACKGROUND

In this chapter, the restoration problem is formulated. The problems associated with restoration are discussed. The approaches, constraints and approximations are outlined. At the end, a brief formulation for three algorithms, to be used in chapter 3 for comparison, are presented.

FORMULATION OF THE RESTORATION PROBLEM

Dealing with continuous images, an original image is represented as a function f(k,l); a blurred image is represented as a function g(x,y); and the PSF of an imagery system would be h(x,y;k,l).

Imagery systems can be modeled as linear systems. Thus, g(x,y), the response of a linear system, is equal to the superpositon of input impulses passing through the system. In mathematical form the imaging equation becomes [1]

$$g(x,y)= \int_{k=-\infty}^{+\infty} \int_{l=-\infty}^{+\infty} f(k,l)\, h(x,y;k,l)\, dk\, dl \quad .$$

Since the PSF is considered as a space-invariant point-spread function (SIPSF), i.e independent of position, then

$$h(x,y;k,l) = h(x-k,y-l) \quad .$$

Thus,

$$g(x,y)= \int_{k=-\infty}^{+\infty} \int_{l=-\infty}^{+\infty} f(k,l)\, h(x-k,y-l)\, dk\, dl$$

which refers to a two-dimensional convolution. Transforming the continuous case to a discrete one, the equation becomes [2]

$$g(x,y)= \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} f(k,l)\, h(x-k,y-l) \quad .$$

Quantizing the function f (the image) gives a two dimensional set of data; in other words it gives a matrix [f]. Also, the PSF h is quantized into a matrix [h] (in case of defocused lens blur) or a sequence (in case of linear motion blur.) The dimension JxK of [h] depends on the number of pixels the blur is extending over. For instance, a defocused lens blur over 6 pixels would result in [h] with 6x6 dimension.

All uncertainities in the restoration process, sensor noise, quantizer noise, any other source of error, or all of the above could be modeled as an additive white Gaussian noise with zero mean and uncorrelated with the orignal image [1,2]. Therefore, considering restoration in its general form, where statistical degradation (noise) is considered in formulation, the equation becomes

$$g(x,y)= \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} f(k,l)\, h(x-k,y-l) \quad + \quad n(x,y)$$

where n(x,y) is the noise term.

Since original images are limited in space, the summation would be finite. Thus,

$$g(x,y)= \sum_{k=1}^{M} \sum_{l=1}^{N} f(k,l)\, h(x-k,y-l) \quad + \quad n(x,y) \qquad (1)$$

where the original image is of size MxN. The lexicographic notation is found to be very appropriate to use here. The lexicographic notation is the notation of stacking the rows of a matrix in a column vector where the rows are stacked in order, i.e [row 1 row 2 . . . . . row M]^T. Using lexicographic or stacking notation in representing original and blurred images, equation (1) becomes [1]

$$\mathbf{g} = [H]\,\mathbf{f} + \mathbf{n} \qquad (2)$$

where **g** is a column vector of (M+J-1) x (N+K-1) elements

f is a column vector of M x N elements

n is a column vector of (M+J-1) x (N+K-1) elements

H is a matrix of (M+J-1)(N+K-1) x (MN) elements.

And this form of the equation is more convenient for usage in restoration.

EXPLORING IMPORTANT FEATURES OF SIPSF [H]

Recall that the blur to be studied in this thesis is of SIPSF type. For mathematical convenience, **f** is appended with zeros to reach the size (M+J-1) x (N+K-1), and H is enlarged to the size (M+J-1)(N+K-1) x (M+J-1)(N+K-1). The resizing of the vector and matrix would not alter any results since it involves the product of zeros (from **f**) and the extra elements in H. By inspection, the enlarged matrix H in equation (2) shows nice features. For linear motion blur, H is

Toeplitz while for defocused lens blur it is block Toeplitz. Both forms are shown below respectively,

$$[H] = \begin{bmatrix} h(0) & & & 0 \\ h(1) & h(0) & & \\ \vdots & \ddots & \ddots & \\ h(M-1) & \cdots & h(1) & h(0) \\ & \ddots & & \ddots \\ 0 & & \ddots & \\ & & h(M-1) & \cdots \quad h(1) \quad h(0) \end{bmatrix}_{P \times P}$$

where $P = M(K+N-1)$,

$$[H] = \begin{bmatrix} [H_0] & & & 0 \\ [H_1] & [H_0] & & \\ \vdots & \ddots & \ddots & \\ [H_{j-1}] & \cdots & [H_1] & [H_0] \\ & \ddots & & \ddots \\ 0 & & \ddots & \\ & & [H_{j-1}] & \cdots \quad [H_1] \quad [H_0] \end{bmatrix}_{Q \times Q}$$

where $Q = (J+M-1)(K+N-1)$,

$[H_j]$ is a Toeplitz matrix of the jth row in the blur matrix h. Therefore, $[H_j]$ looks like

$$[H_j]=\begin{bmatrix} h(j,0) & & & & \\ h(j,1) & h(j,0) & & & \\ \vdots & \ddots & \ddots & & \\ h(j,K-1) & \cdots & h(j,1) & h(j,0) & \\ & \ddots & & \ddots & \ddots \\ 0 & & \ddots & & h(j,K-1) \cdots h(j,1)\ h(j,0) \end{bmatrix}_{L \times L}$$

where L = K+N-1.

These forms of SIPSF offer more efficient use of computer memory. Storing the complete form of [H] is not practical due to limited physical memory on a computer. The repetition of the non-zero elements and their concentration in the middle diagonals lower the number of elements to be stored.

Another feature of the Toeplitz form of [H] can be utilized. It has been proven that a Teoplitz matrix and a circulant matrix are asymptoticaly equivalent [1]. Thus, within a small error, a Toeplitz matrix can be substituted by a circulant matrix. The rows of a circulant matrix are the cyclic shift of each other. Taking the Toeplitz matrix of the linear motion case, the circulant matrix $[H_c]$ becomes

$$[H_c]=\begin{bmatrix} h(0) & & & & h(M-1) & \cdot & \cdot & h(1) \\ h(1) & h(0) & \cdot & & & h(M-1) & \cdot & h(2) \\ \cdot & & \cdot & & & & & \\ \cdot & & & \cdot & & & & \\ h(M-1) & \cdot\cdot & h(1) & h(0) & \cdot & & & \\ & \cdot & & & & & \cdot & \\ & & \cdot & & & & & \\ & & & & h(M-1) & \cdot\cdot & h(1) & h(0) \end{bmatrix}_{P \times P}$$

where P = M(K+N-1).

However, it is known that the eigenvalues of a circulant matrix are the Discrete Fourier Transform (DFT) of its cyclic sequence [1]. Thus, eigenvalues of $[H_c]$ = DFT of [h(0) . . . . . h(M-1)].

The same argument could be extended to the defocused lens case [3] where [H] is a block Toeplitz; and, therefore, it is approximately equivalent to a block circulant matrix. The eigenvalues of such a matrix would be the two-dimensional Discrete Fourier Transform (2-D DFT) of the blur matrix $[h]_{JxK}$.

These two features are going to be utilized in the pseudo-inverse algorithms that follow.

PROBLEMS OF RESTORATION

Again, the general equation for restoraton is

$$\mathbf{g} = [H]\,\mathbf{f} + \mathbf{n}$$

solving for **f**, the direct inverse technique (deconvolution) would be

$$[H]^{-1}\mathbf{g} = [H]^{-1}[H]\,\mathbf{f} + [H]^{-1}\,\mathbf{n}$$

and $\mathbf{f} = [H]^{-1}\mathbf{g} - [H]^{-1}\mathbf{n}$.

There are two main problems with this approach [1]. These are (1) the singularity of the blur matrix (It is common in digital image restoration to have blurs where their $[H]^{-1}$ does not exist.) and (2) the ill-conditioning nature of the restoration problem. (This is when a small perturbation in data (noise) would be amplified by large-valued elements in $[H]^{-1}$. This problem is most frequent.)

In trying to regularize the restoration problem, a new problem is generated, namely, ringing artifacts. These are defined as periodic overshoots and undershoots about an edge that decay in spatial coordinates as we move further from the edge. The cause of these artifacts is the deviation of the regularized filter from the direct inverse [4].

APPROACHES FOR RESTORATION

The restoration problem has been approached in many different ways. In all the approaches some assumptions, approximations, and constraints are made. Assumptions are made in order to present a model which is appropriate for mathematical manipulation while approximations are done to ease the load of computations, to ease the solution of equations, and to build practical algorithms. Always in solving the restoration problem a constraint or criterion is imposed. When this criterion is reached, the restoration is achieved. Different approaches might have different criteria. When the criterion is achieved, then the solution is optimal in the mathematical sense but not necessarily in the visual sense.

The reason is due to our incomplete understanding of the vision process and the simplicity of the criterion in terms of mathematical manipulation [2].

In this work, only three approaches are to be formulated. These are the pseudo-inverse [1], the neural network [5], and the regularized iterative methods [4].

PSEUDO-INVERSE FILTER

Formulation of this filter is done without considering the noise term in equation (2), i.e $\mathbf{n} = [0]$, thus

$$\mathbf{g} = [H]\,\mathbf{f}\,.$$

Using the circulant approximation, we get

$$\mathbf{g} \simeq [H_c]\,\mathbf{f}\,.$$

Representing $[H_c]$ in its diagonal form ,

$$[H_c] = [f]\,[\Lambda_h]\,[f]^{-1}$$

where $[\Lambda_h]$ is a diagonal matrix of the eigenvalues of $[H_c]$ and $[f]$ and $[f]^{-1}$ are Fourier Transform operators.

Thus, $\mathbf{g} \simeq ([f]\,[\Lambda_h]\,[f]^{-1})\,\mathbf{f}$

or $$[f]^{-1}\,\mathbf{g} \simeq [\Lambda_h]\,[f]^{-1}\,\mathbf{f}\,. \qquad (3)$$

The two terms $[f]^{-1}\,\mathbf{f}$ and $[f]^{-1}\,\mathbf{g}$ are equivalent to the DFT (or 2-D DFT) of the vectors f and g, respectively. Equation (3) then becomes

$$G \simeq [\Lambda_h]\;F$$

where G and F are the DFT's (or 2-D DFT's) of **g** and **f**, respectively, and $[\Lambda_h]$ equals the DFT (or 2-D DFT) of the blur sequence (or matrix) as shown in the features of SIPSF section. Solving for F, we get

$$F \simeq [\Lambda_h]^{-1} G .$$

Now, if the matrix [H] is singular, some of its eigenvalues are zeros, and the corresponding elements of $[\Lambda_h]^{-1}$ do not exist because they are the result of dividing by zero. To avoid dividing by zero, the pseudo-inverse of [H] is used to substitute the direct inverse ($[H]^{-1}$). Consequently, any element of $[\Lambda_h]^{-1}$ which is a division by zero would be replaced by zero.

In summary, the steps of the algorithm are as follows.

1. Compute G, the DFT of the blurred image (as one sequence) in case of linear motion blurring or the 2-D DFT of the blurred image (as a matrix) in case of defocused lens blurring.
2. Compute B, the DFT of the blurring sequence in case of linear motion blurring, or the 2-D DFT in case of defocused lens blurring. The blur sequence or matrix should be appended with zeros to reach the size of the blurred image. Note that the elements of B are equal to the elements of the diagonal of $[\Lambda_h]$.
3. Divide G by B, an element by element division except for zero values of B where the division output is replaced by zeros.
4. Compute the inverse DFT (or 2-D DFT) on the output of step 3.

NEURAL NETWORK APPROACH

In this approach, a neural network is used to represent an original image. Each pixel is represented by a cluster of neurons. The size of the cluster equals the value of the gray level of the corresponding pixel. The neurons are mutually interconnected, and self-feedback is allowed.

The energy function of such network is

$$E = -\frac{1}{2}\sum_{i=1}^{L^2}\sum_{j=1}^{L^2}\sum_{k=1}^{M}\sum_{l=1}^{M} T_{i,k;j,l}\, v_{i,k}\, v_{j,l} - \sum_{i=1}^{L^2}\sum_{k=1}^{M} I_{i,k}\, v_{i,k}$$

where M is the maximum gray level in the image,

L^2 is the total number of pixels,

$T_{i,k;j,l}$ is the interconnection strength between neurons (i,k) and (j.l),

$v_{i,k}$ is the binary output of neuron (i,k), and

$I_{i,k}$ is the bias input of neuron (i,k) .

However, the restoration problem can be formulated as one of minimizing an error function, with constraints, defined as

$$E = \frac{1}{2} \| \mathbf{g} - [H] \hat{\mathbf{f}} \|^2 + \frac{1}{2} \lambda \| [D] \hat{\mathbf{f}} \|^2$$

where $\hat{\mathbf{f}}$ is the estimated image

λ is a relaxation constant, and

D is a Laplace operator.

After some mathematical manipulation and equating the two equations, we get

$$T_{i,k;j,l} = -\sum_{p=1}^{L^2} h_{p,i}\, h_{p,j} - \lambda \sum_{p=1}^{L^2} d_{p,i}\, d_{p,j} \qquad (4)$$

and

$$I_{i,k} = \sum_{p=1}^{L^2} g_p\, h_{p,i} \qquad (5)$$

where $h_{p,i}$ and $d_{p,i}$ are the elements of the matrices H and D respectively. Moreover, after further mathematical manipulation, we get

$$u_{i,k} = \sum_{j=1}^{L^2} T_{i,.;j,.} \quad g_j \quad + \quad I_i \tag{6}$$

where $u_{i,k}$ is the actual output of neuron (i,k), and

$$\Delta v_{i,k} = \begin{cases} \Delta v_{i,k} = 0 & \text{if } u_{i,k} = 0 \\ \Delta v_{i,k} = 1 & \text{if } u_{i,k} > 0 \\ \Delta v_{i,k} = -1 & \text{if } u_{i,k} < 0 \end{cases} \tag{7}$$

$$g_i^{new} = \begin{cases} g_i^{old} + \Delta v_{i,k} & \text{if } \Delta E < 0 \\ g_i^{old} & \text{if } \Delta E \geq 0 \end{cases} . \tag{8}$$

Here ΔE is the change in energy which is calculated as

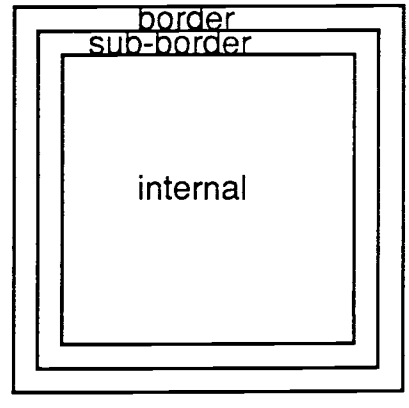
$$\Delta E = - u_{i,k} \Delta v_{i,k} - \frac{1}{2} T_{i,.;i,.} (\Delta v_{i,k})^2 . \tag{9}$$

Putting the algorithm in words, it would be:

1. Compute the interconnection strengths T and bias inputs I for all neurons as shown in equations (4) and (5), respectively.
2. Take the degraded image as the initial estimation.
3. For pixel g_i use equations (6), (7) and (9) to calculate the actual output, the change in the binary output, and the change in energy, respectively.
4. Update the pixel g_i, using equation (8). Repeat step 3 and 4 until $\Delta E \geq 0$ or $v_{i,k} = 0$.
5. Visit the next pixel and start again from step 3 until all the pixels are visited.

6. Check the energy function. If energy does not change, a restored image is obtained; otherwise, go to step 3 for another iteration.

Due to approximations in the solution, ringing artifacts are generated. To suppress such artifacts, the following scheme is suggested. The image is divided into 3 regions: namely, internal, sub-border and border regions. For the border region, the pixels must be kept equal to those in the degraded image. λ in equation (4) could be given different values for different regions; or alternatively, λ could be set to zero in case of high signal to noise ratio (SNR), and another parameter T is used. The difference between the degraded pixel and the restored one should not exceed a certain value of T in the sub region The regions of an image are shown below.



REGULARIZED ITERATIVE APPROACH

Although the blur considered is SIPSF, this approach would deal with it as a space-variant point-spread function (SVPSF). The variation of PSF depends on local statistical characteristics.

In this approach, two criteria and constraints are used. These are:

1. $[(\mathbf{g} - [H]\hat{\mathbf{f}})^t R (\mathbf{g} - [H]\hat{\mathbf{f}})]^{\frac{1}{2}} \leq \varepsilon_1$,

where R is a diagonal matrix containing weight coefficients, ranges between 0 and 1, for all the picture elements. R is a local effect on PSF.

2. $[(D\hat{\mathbf{f}})^t S (D\hat{\mathbf{f}})]^{\frac{1}{2}} \leq \varepsilon_2$,

here S is a diagonal weighting matrix containing positive coefficients in the range (0,1). S is another local effect on PSF.

The proposed solution for this approach is the Miller regularization solution $\hat{\mathbf{f}}_m$ that minimizes the following function,

$$\Phi(\hat{\mathbf{f}}) = [(\mathbf{g} - [H]\hat{\mathbf{f}})^t R (\mathbf{g} - [H]\hat{\mathbf{f}})]^{\frac{1}{2}} + \alpha [(D\hat{\mathbf{f}})^t S (D\hat{\mathbf{f}})]^{\frac{1}{2}} .$$

The solution would be

$$(H^t R H + \alpha D^t S D)\hat{\mathbf{f}}_m = H R \mathbf{g} .$$

To reduce the computational complexity of the above equation, the solution $\hat{\mathbf{f}}_m$ is approximated, and the solution would be

$$\hat{\mathbf{f}}_m = (I - \alpha \beta D^t S D)\hat{\mathbf{f}}_m + \beta H^t R (\mathbf{g} - H\hat{\mathbf{f}}_m)$$

where α is a regulariztion parameter,

β is a relaxation parameter, and

I is an identity matrix .

In an iterative scheme the solution could be written as

$$\hat{f}_{k+1} = (I - \alpha \beta D^t S D)\, \hat{f}_k + \beta H^t R (g - H \hat{f}_k) .$$

Using a priori knowldge as a constraint on the solution, the projection operator P is introduced in every iteration, and the restored pixels are not allowed to exceed specific bounds. Then, the final solution would be

$$\hat{f}_{k+1} = P[(I - \alpha \beta D^t S D)\, \hat{f}_k + \beta H^t R (g - H \hat{f}_k)]. \qquad (10)$$

The first term in equation (10) performs a low pass filtering action; the activity of this filter is controlled by the weights in S. The second term in the equation regulates the size of the restoration term $\beta H^t R (g - H \hat{f}_m)$, or in effect, it varies the relaxation parameter β.

Determination of R depends on the degree of image degradation. For moderate degradation, R could be used as an identity matrix. On the other hand, to calculate S the following equations are to be used.

$$\sigma_g^2(i,j) = \frac{1}{(2P+1)(2Q+1)} \sum_{k=i-P}^{i+P} \sum_{l=j-Q}^{j+Q} [\, g(k,l) - m_g(i,j)]^2 \qquad (11)$$

where σ_g^2 is the local variance in the region of pixel(i,j),

(2P+1)(2Q+1) is the size of the analysis window,

$m_g(i,j)$ is the local mean of the region of pixel(i,j), and

$$m_g(i,j) = \frac{1}{(2P+1)(2Q+1)} \sum_{k=i-P}^{i+P} \sum_{l=j-Q}^{j+Q} g(k,l) \quad . \tag{12}$$

This leads to the calculation of $s_{i,j}$ using the following equation.

$$s_{i,j} = \frac{1}{1 + \mu \ \mathrm{Max}\ [0 \ , \ \sigma_g^2(i,j) - \sigma_n^2]} \tag{13}$$

where σ_n^2 is the noise variance, and

μ is a tuning parameter.

In summary, the algorithm steps are:

1. Use initial value $\hat{f}_0 = g$.
2. Compute S and R matrices using equations (12), (11) and (13).
3. Use equation (10) iteratively until no significant change is observed.

3. PROPOSED ALGORITHM AND SIMULATIONS

In this chapter I propose a new algorithm to solve the singularity problem in a noise-free environment. At the end of this chapter, computer simulations of my algorithm, the pseudo-inverse filter, the neural network approach and the regularized iterative technique are performed. The simulations are done for noise-free and noisy environments.

PROBLEM STATEMENT

As mentioned earlier, two main problems are involved in the restoration process when using direct deconvolution. The first one is the amplification of noise, and the second is the singularity in blur matrices.

In this work a solution for a noise-free environment is sought. Later the effect of noise on such solution will be studied. Thus, the restoration equation

$$\mathbf{g} = [H]\,\mathbf{f} + \mathbf{n}$$

becomes

$$\mathbf{g} = [H]\,\mathbf{f}$$

and the only problem left is the singularity of [H], i.e, when the inverse of [H] does not exist. This occurs when the Fourier Transform of [h] contains zeros. Facing this problem, researchers dropped the deconvolution approach and used estimation and optimization theories to solve the problem approximately.

SINGULARITIES OF UNIFORM BLUR MATRICES

It is found, empirically, for any matrix having its halves as mirror images of each other that the 2-D DFT of such a matrix will have zeros as elements in its middle row, middle column or both. The particular form depends on the normal direction of the virtual mirror. The above statement is true if the number of elements in the normal direction, in the space domain, is even. As an example, consider the following matrix which has mirror image structure in the column direction

$$\begin{bmatrix} 1 & 2 & 2 & 1 \\ 3 & 4 & 4 & 3 \\ 5 & 6 & 6 & 5 \\ 7 & 8 & 8 & 7 \end{bmatrix}$$ it's 2-D 4x4 DFT will be $$\begin{bmatrix} - & - & 0 & - \\ - & - & 0 & - \\ - & - & 0 & - \\ - & - & 0 & - \end{bmatrix}$$

It is also true that if any of these halves consist of mirror images and exhibit the same conditions as mentioned above, then columns (or rows) of zeros will exist in the middle of the corresponding halves as described above.

A uniform blur matrix with even numbers of columns and rows exhibits such structure of singularity in both directions.

For uniform linear matrix blur, the discrete PSF could be approximated as [2]

$$h(k,l) = \frac{1}{(l_2+1)} \qquad \text{for } l = 0,- - - -, l_2 \text{ and } k = 0$$

$$= 0 \qquad \text{elsewhere.}$$

l_2 is the number of pixels over which the motion is extended. For a uniform linear motion blurring over 6 pixels, the discrete PSF would be

$$h(k,l) = \frac{1}{6+1} = 0.1428 \qquad \text{over 6 pixels,}$$

yielding a blur matrix $[h]_{1x6}$ (i.e, a sequence) as

$$[h] = [\ 0.1428\ 0.1428\ 0.1428\ 0.1428\ 0.1428\ 0.1428\ ]\ .$$

As indicated earlier, since it is uniform and has an even number of elements, this blur matrix is singular. Thus, its Fourier Transform sequence has a zero at its middle as shown here

$$[- - -\ 0\ - -]\ .$$

Fig.1 shows the blur matrix in the Fourier domain.

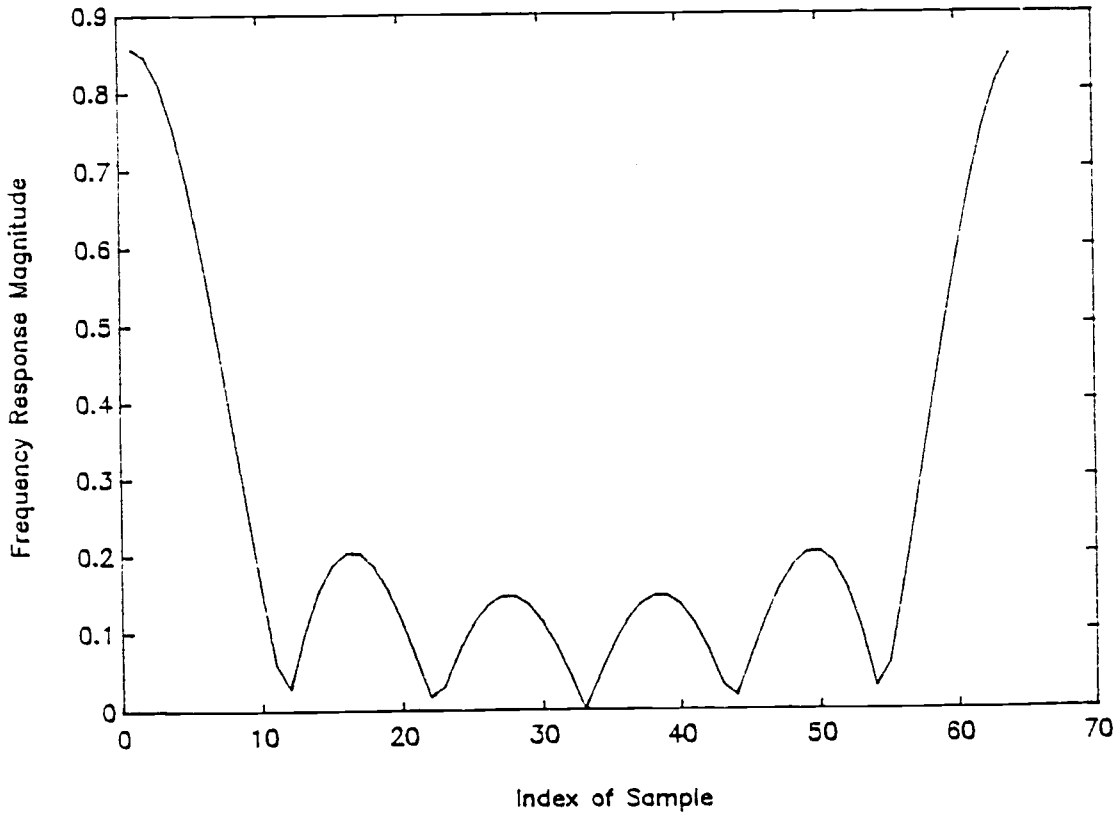


Fig. 1 The PSF of linear motion blur in the frequency domain.

For a uniformly defocused lens with circular aperture, the PSF is approximated by [2]

$$h(k,l) = \frac{1}{\pi R^2} \quad \text{for } \sqrt{k^2+l^2} \leq R,$$

$$= 0 \quad \text{elsewhere,}$$

where R is the radius of an imaginary cylinder approximating the PSF.

For a uniformly defocused lens blurring over 6 pixels, I choose the discrete PSF to be approximated as

$$h(k,l) = \frac{1}{35} = 0.02857 \quad \text{over a grid of 6x6 pixels.}$$

This yields a blur matrix $[h]_{6x6}$, as

$$\begin{bmatrix} 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & .02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \end{bmatrix}.$$

As indicated earlier, this uniform blur matrix is singular with mirror image structure in 2 dimensions. Thus, its Fourier Transform matrix has its middle column and row full of zeros as shown below.

$$\begin{bmatrix} - & - & - & 0 & - & - \\ - & - & - & 0 & - & - \\ - & - & - & 0 & - & - \\ 0 & 0 & 0 & 0 & 0 & 0 \\ - & - & - & 0 & - & - \\ - & - & - & 0 & - & - \end{bmatrix}$$

Fig.2 shows the blur matrix in the Fourier domain.

Both of these uniform blurs will be used later in computer simulations.

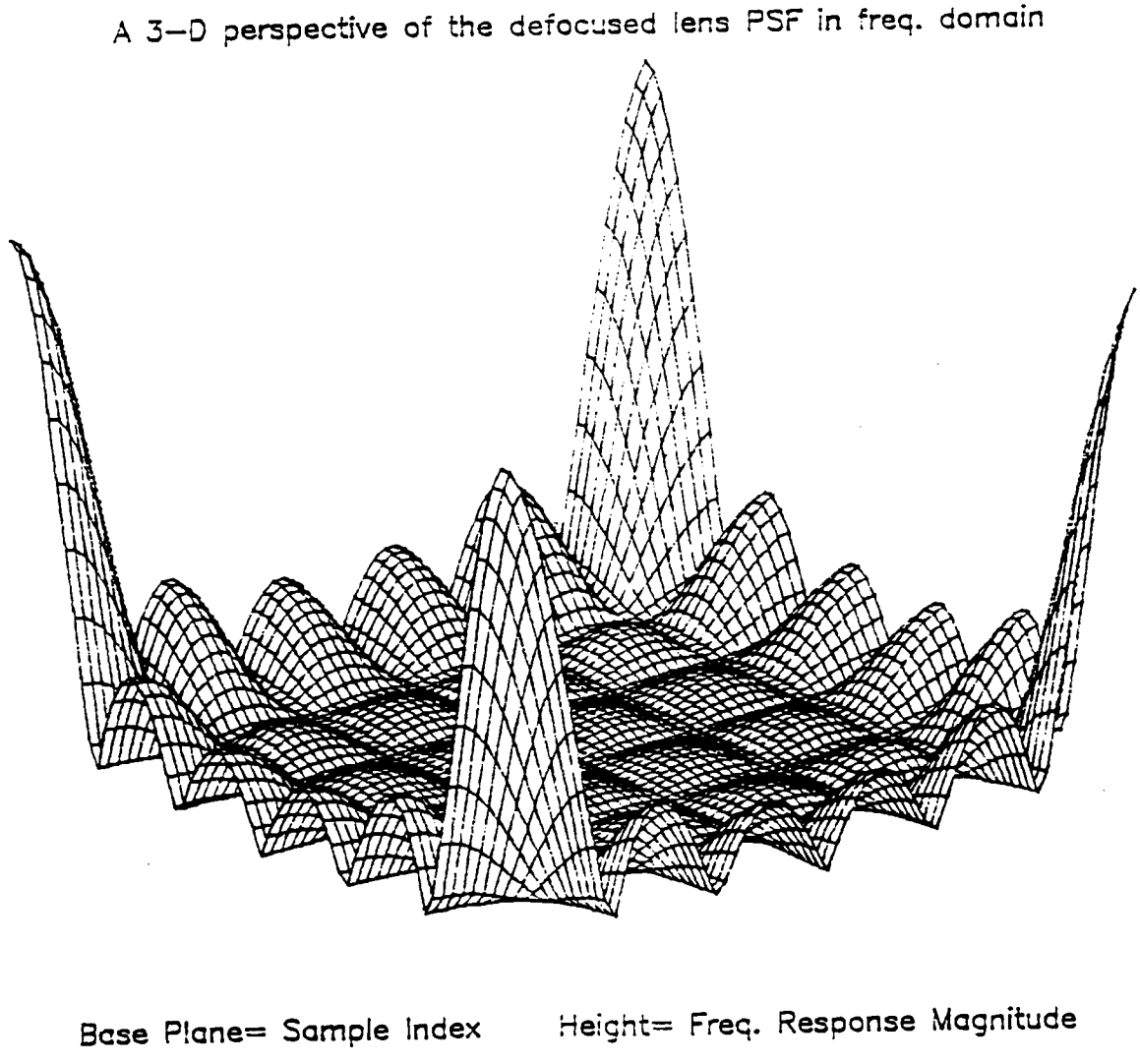


Fig. 2 The PSF of defocused lens blur in the 3-dimensional frequency domain.

MY APPROACH TO THE SOLUTION

Blurring by a SIPSF blur is, simply, an element by element product of the original image and the blur matrix in the frequency domain. Thus, the singular elements in the blur matrix are transfered to the blurred matrix. In other words, the number of zeros, comprising the singular elements, in both matrices are equal. Moreover, these zeros, as shown in the previous section, form only a few strips in the frequency domain, i.e they do not dominate the spectrum.

Keeping this in mind and recalling that the deconvolution process is an element by element division of the blurred image and the blur matrix in the frequency domain leads us to the following conclusion: the output of such division possesses most of the information (energy) about the original image. The only missing information is where the singularity occurs (forced zero elements). Thus, the pseudo-inverse solution (in the frequency domain) lies in the neighborhood of the exact solution. It needs only some kind of an iteration method that will converge the approximate solution to the exact solution through well-known constraints.

The iteration I am proposing travels between frequency domain and space domain with two constraints. The non-singular elements (in frequency) of the output of the pseudo-inverse approach comprise a perfect constraint for a converging solution. Another constraint, in the space domain, would be to impose appended zeros on the solution when transforming the output of the division to the space domain after each iteration.

STEPS OF THE ALGORITHM

I found that the following heuristic algorithm, utilizing the above constraints, would do the job:

1. Append zeros to the blurred image until the numbers of columns (and rows in case of defocused lens blur) are twice that of the original image matrix.
2. Compute the DFT (2-D DFT) of the appended image in case of linear motion blur (defocused lens blur.)
3. Append zeros to the blur matrix until it has the same size as the appended blurred image matrix.
4. Compute the DFT (2-D DFT) of the appended blur sequence (matrix) in case of linear motion blur (defocused lens blur.)
5. Perform an element by element division of the output of step 2 by the output of step 4. If the denominator is zero, replace the division output by zero (singular elements.)
6. Compute the inverse DFT (2-D DFT) of the output of the division in case of linear motion blur (defocused lens blur.)
7. Chop the extra rows and columns of the output of step 6 back to the size of the original image.
8. Append zeros to the output of the chopped matrix until the size of the matrix is doubled.
9. Compute the DFT (2-D DFT) of the output of step 8 in case of linear motion blur (defocused lens blur.)
10. Replace all the elements of the output of step 9 by the elements of the output of step 5 except for the singular elements of step 5.
11. Compute the inverse DFT (2-D DFT) of the output of step 10 and

chop the extra rows and columns of the outcome.

12. Repeat step 8 through 11 until the restored image is satisfactory.

FLOW CHART OF MY ALGORITHM

For simplicity in the flow chart, the acronyms LMB and DLB are used. They stand for linear motion blur and defocused lens blur, respectively. Also, the original image size is assumed to be MxN. The flow chart is shown in Fig.3 in the next page.

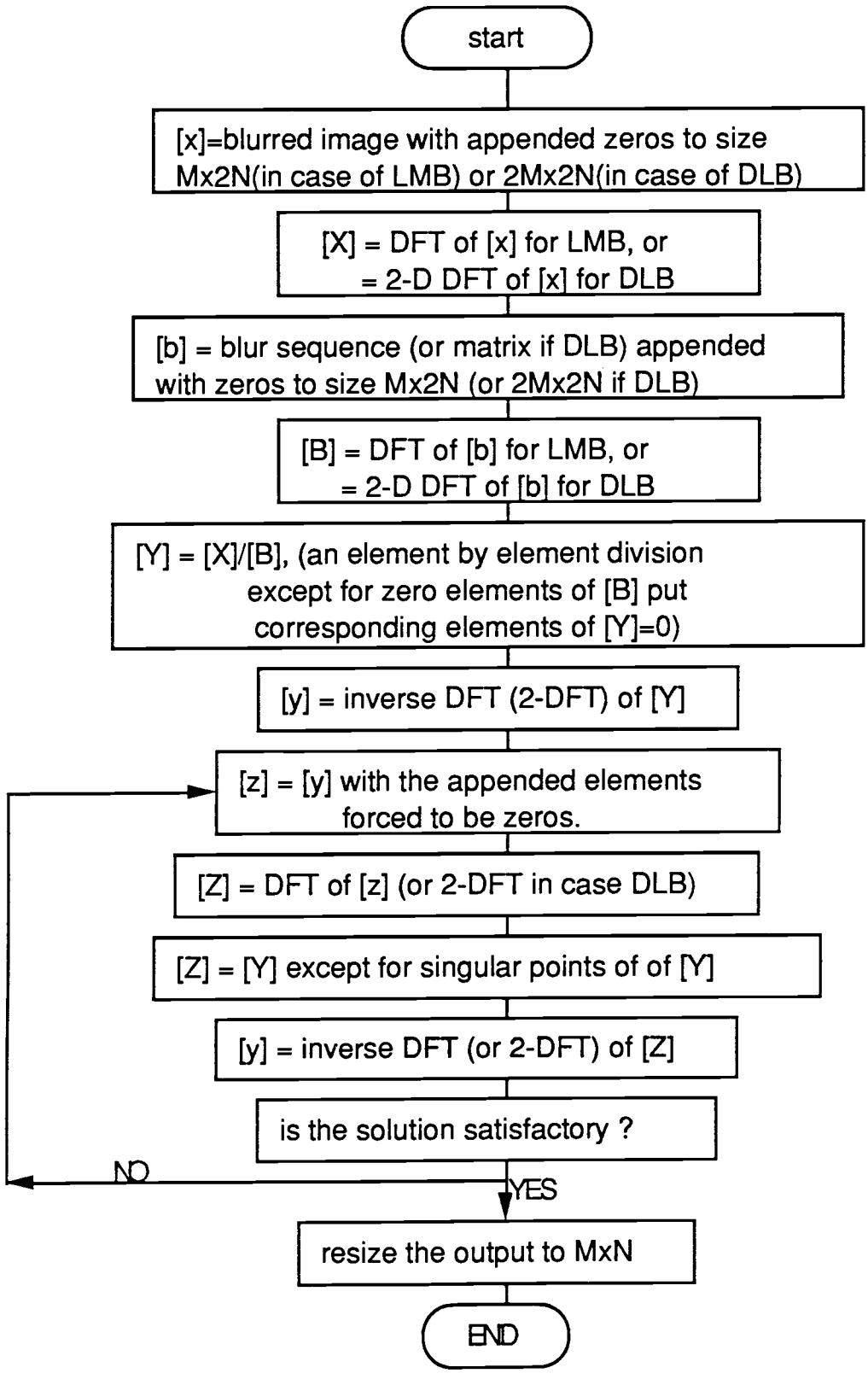


Fig.3 Flow chart of my algorithm.

COMPUTER SIMULATIONS

My algorithm was tested against the pseudo-inverse, neural network, and regularized iterative algorithms. The environment was first considered noise-free; then noise was introduced and some simulations are repeated.

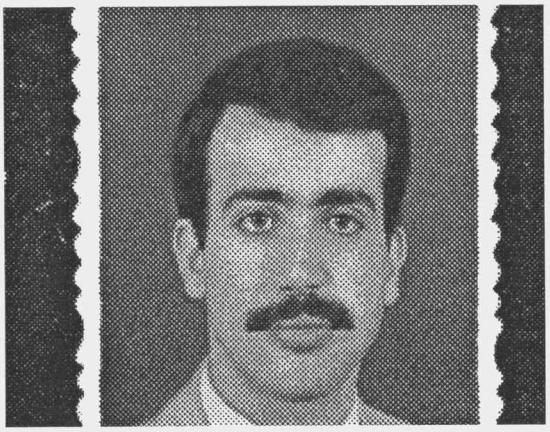
In the simulations an original image called "ahmad" and shown in Fig.4a is used. It consists of 256x200 pixels. The gray level of each pixel is represented by 5 bits, i.e, the gray level value is between 0 and 31. The image "ahmad" is artificially blurred by linear motion over 6 pixels with PSF

$$[h] = [\ 0.1428\ \ 0.1428\ \ 0.1428\ \ 0.1428\ \ 0.1428\ \ 0.1428\ ]\ .$$

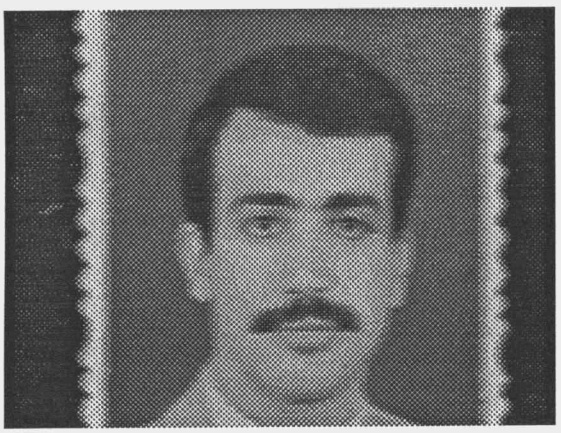
The linear motion blurred image is shown in Fig.4b. Again, the image "ahmad" is artificially blurred by a defocused lens over a grid of 6x6 pixels with PSF [h] equal to

$$\begin{bmatrix} 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & .02857 & 0.02857 & 0.02857 & 0.02857 \\ 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 & 0.02857 \end{bmatrix}.$$

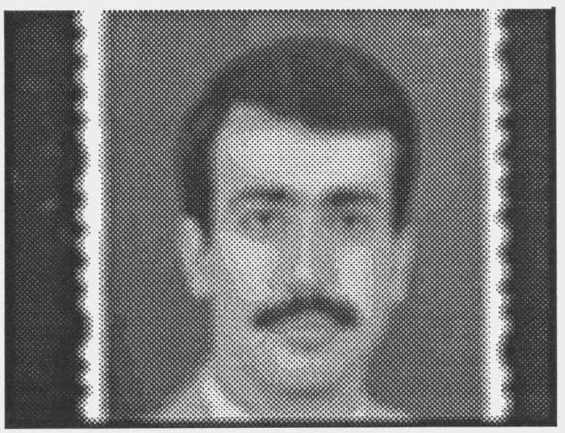
The defocused lens blurred image is shown in Fig.4c. Blurring and restorations were performed using Tektronix 4319 and 4315 workstations (computers).



a)



b)



c)

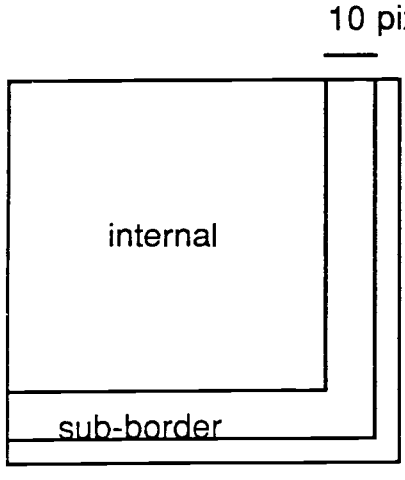
Fig. 4 a) upper : the original image "ahmad",

b) lower left : the linear motion blurred image,

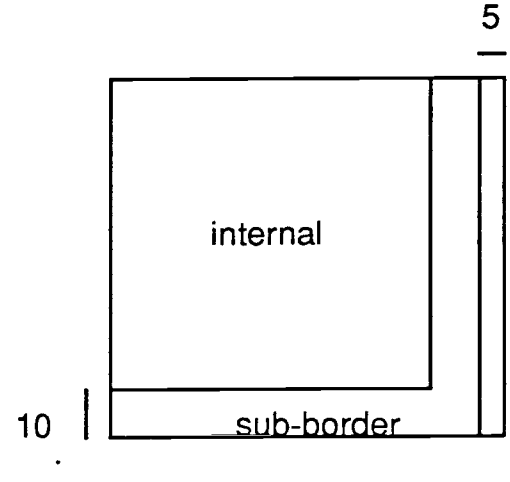
c) lower right : the defocused lens blurred image.

The algorithms used for restoration are as described in the previous chapter. Using the pseudo-inverse filter, the restored image from linear motion blurring is shown in Fig.5a while the restored image from defocused lens blur is shown in Fig.6a.

Using the neural network approach, the restored image from linear motion blurring is shown in Fig.5b while the restored image from defocused lens blurring is shown in Fig.6b. Twenty iterations were needed for linear motion blur while fourty iterations were used for the defocused lens case. To reduce ringing artifacts, the border region and the sub-border region were taken as 5 pixels and 10 pixels wide respectively. The parameter T equals 1.0. Note that in my simulations, the image partitioning suggested by reference [5] does not suppress ringing artifacts as well as the following partitioning.



Defocused Lens Case



linear motion case

Using the regularized iterative technique, the restored image from linear motion blurring is shown in Fig.5c while the restored

image from defocused lens blurring is shown in Fig.6c. Twenty iterations were needed for the linear motion case, and fourty iterations were used for the defocused lens case. The parameters used were α=0.001, β=1.91, μ=0.5, and σ_n^2= 0.

Concerning my algorithm, the necessity for utilizing a Fast Fourier Transform (FFT) algorithm is obvious. In order to reduce restoraton time, the split radix FFT is used as described in reference [6]. This FFT algorithm is reported to be the fastest. Using my algorithm, the restored image from linear motion blurring is shown in Fig.5d while the restored image from defocused lens blurring is shown in Fig.6d. For the defocused lens case, fifteen iterations were needed. It took about one hour. However, it was observed that after a few iterations, the converging pixels showed a repetitive pattern. In an odd numbered row, an odd numbered pixel in the converging image exceeded its corresponding pixel in the original image by a constant value while an even numbered pixel fell behind its corresponding pixel by the same constant value. For an even numbered row, the opposite was true. For an odd numbered row, the constant value was found to be

constant value = last pixel in the row - diff where

diff = pixel(1,1) in the original - pixel(1,1) in the converging image.

For an even numbered row, the constant value was

constant value = last pixel in the row + diff .

The last pixel in a row is used in the constant value because it is supposed to be equal to zero in the enlarged original image.

To speed up the converging process, the above values were computed for each row and added or subtracted accordingly. Using the speedup scheme, two iterations were needed for the linear motion blur case, and eight iterations were needed for the defocused lens blur case. For the defocused lens blur case, Fig.7 shows the restored images after each iteration.

RESTORING NOISY IMAGES

Noise is added to the defocused lens blurred image shown in Fig.4a. The noisy blurred image with 50 dB signal to noise ratio is shown in Fig.8a.

The neural network algorithm, the regularized iterative algorithm and my algorithm were used on the noisy blurred image. Since the pseudo-inverse did not work for me in the noise-free environment, there is no need to use it in noisy environments. The neural network and the regularized iterative algorithms showed robustness to noise. Their restorations are shown in Fig.8b and Fig.8c, respectively. My algorithm was overwhelmed by the added noise and failed to produce any comprehensible image.

QUANTITATIVE RESULTS

More quantitative results, used for performance comparisons, are shown in the next chapter. Tables 1, 2 and 3 show the consumption of time and the error introduced in the restoration process by the different algorithms.

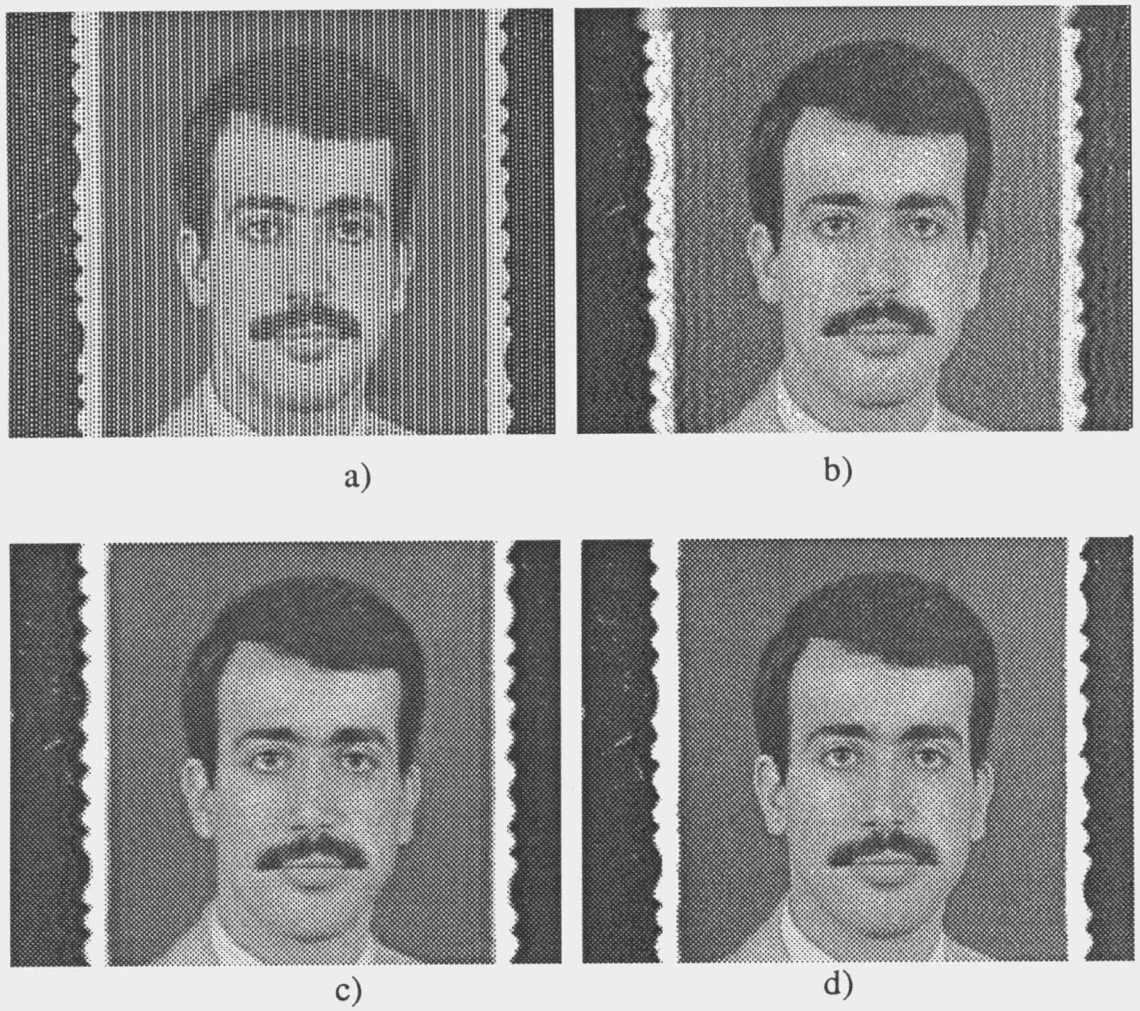


Fig. 5 Linear motion case, the restored images,

a) upper left : using pseudo-inverse filter,

b) upper right : using neural network approach,

c) lower left : using regularized iterative approach,

d) lower right : using my algorithm.

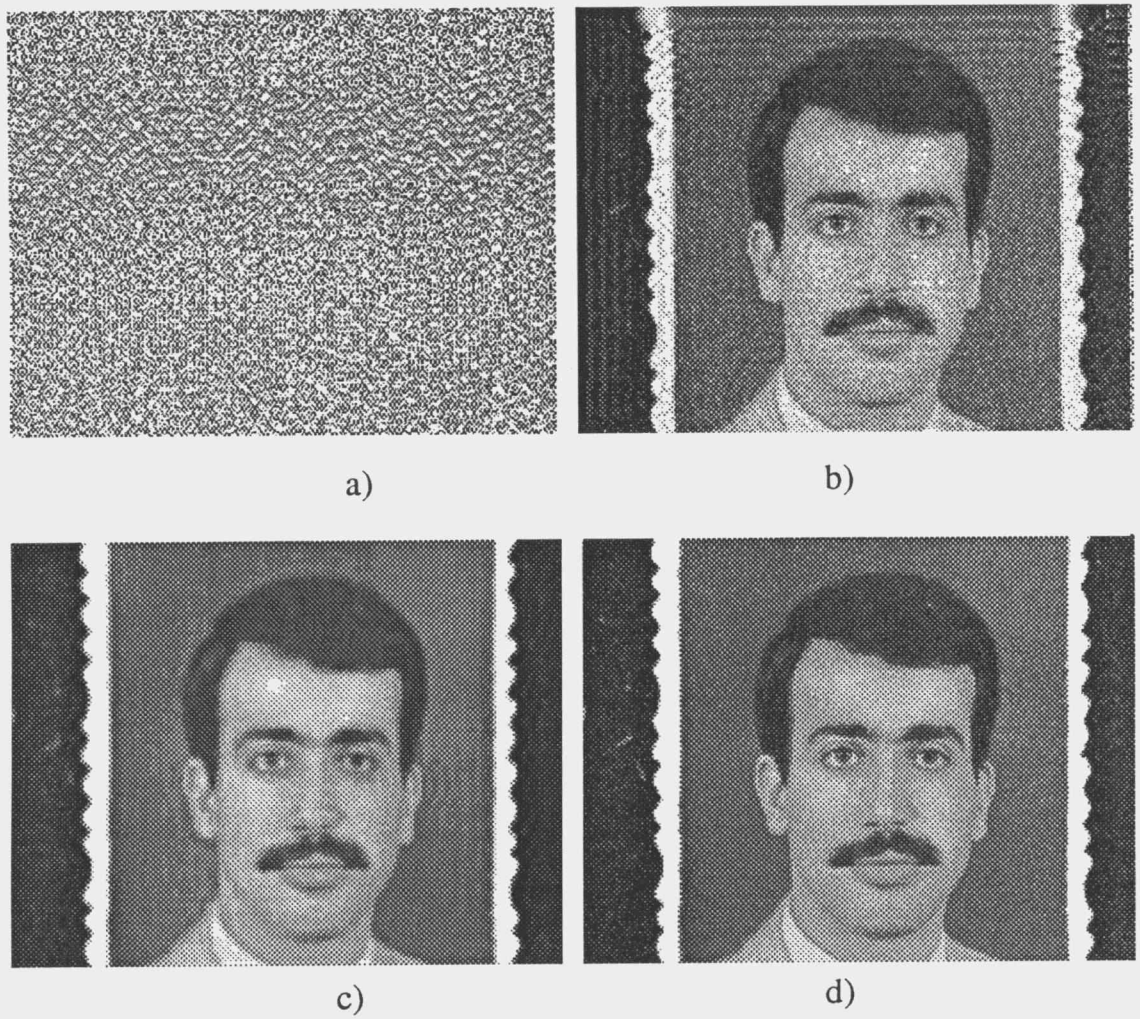


Fig. 6 Defocused lens case, the restored images,

a) upper left : using pseudo-inverse filter,

b) upper right : using neural network approach,

c) lower left : using regularized iterative approach,

d) lower right : using my algorithm.

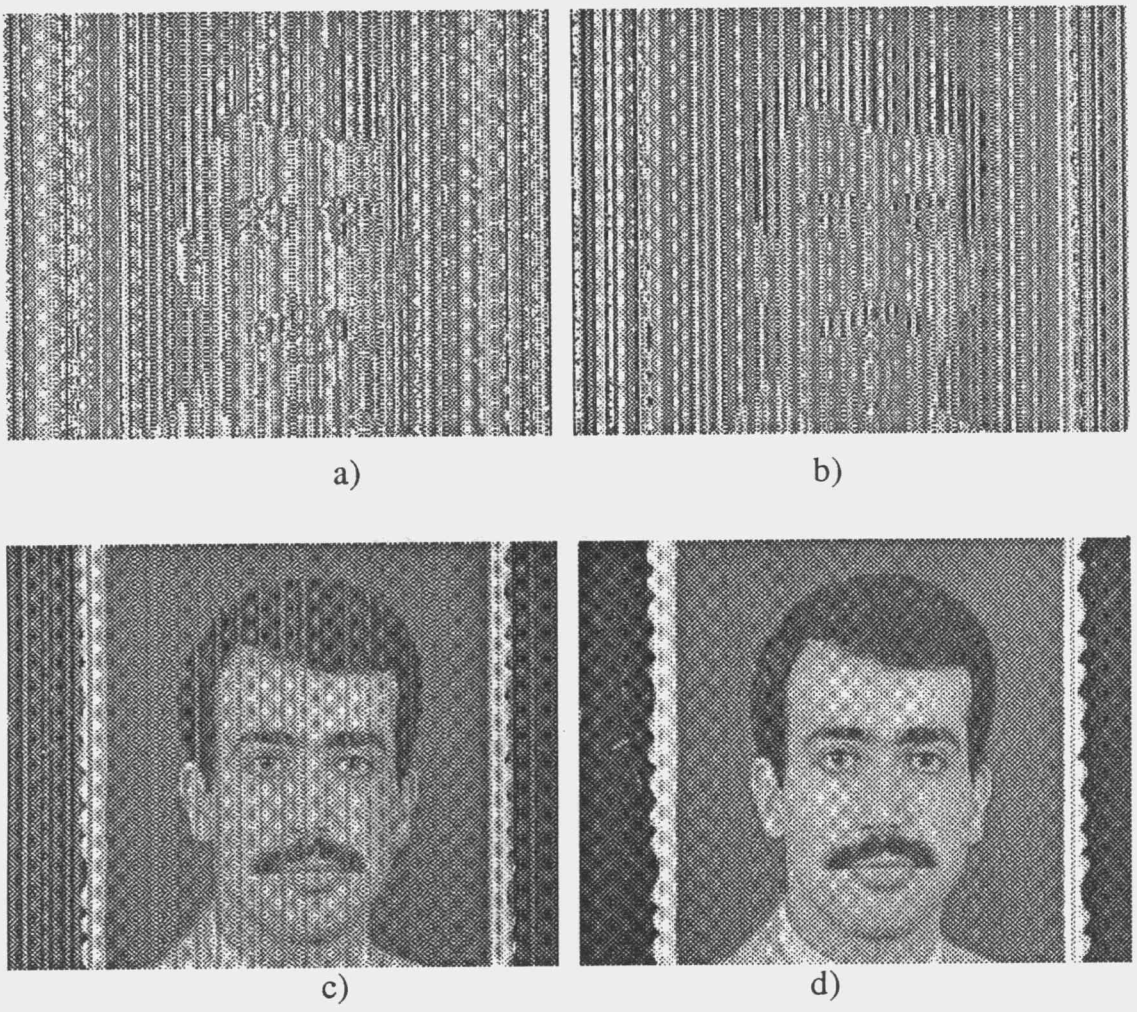


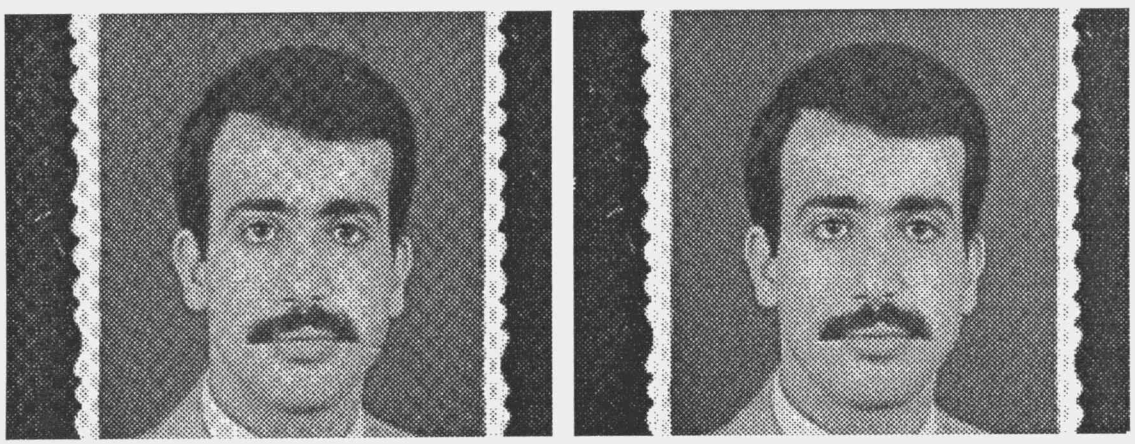
Fig. 7 The restored images using my algorithm,

a) upper left : after 2 iterations,

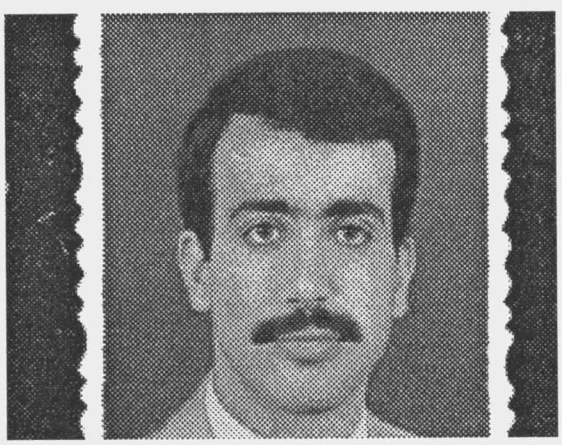
b) upper right : after 3 iterations,

c) lower left : after 4 iterations,

d) lower right : after 5 iterations.



e) f)



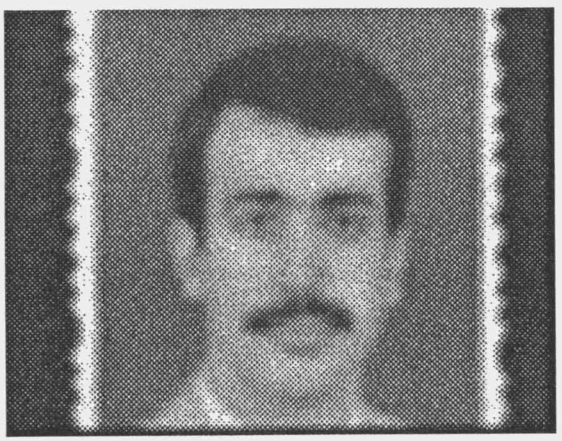
g)

Fig. 7 (continued) The restored images using my algorithm,

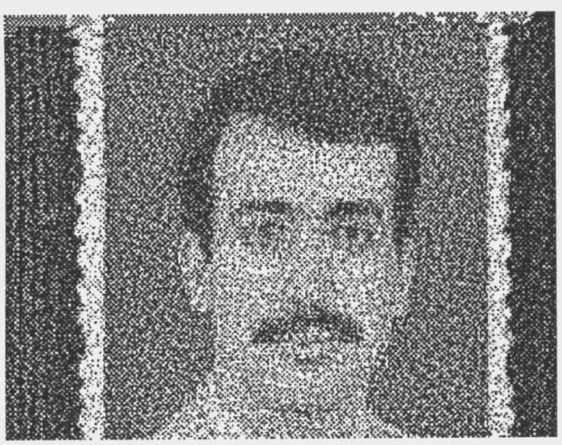
e) upper left : after 6 iterations,

f) upper right : after 7 iterations,

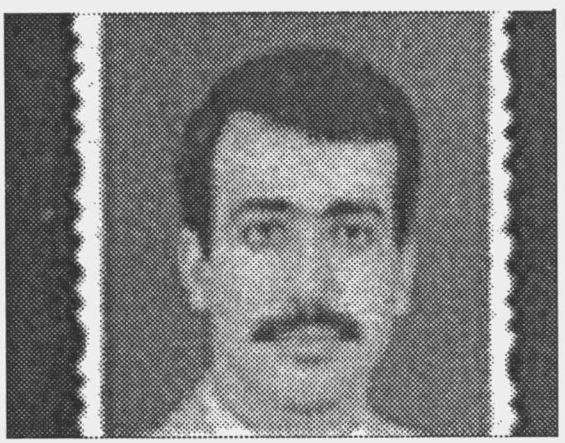
g) lower : after 8 iterations.



a)



b)



c)

Fig. 8 Noisy images,

a) upper : noisy defocused lens blurred image (SNR=50 dB),

b) lower left: restored image using neural network approach,

c) lower right: restored image using regularized iterative approach.

4. DISCUSSION

In this chapter some performance criteria are discussed. Also, the results of the simulations are quantitized and compared. Advantages and disadvantages of each algorithm are emphasized.

PERFORMANCE CRITERIA

Two criteria are adopted by most researchers to evaluate performance [2]. The first is visual quality. Looking at a restored image with the naked eye would indicate how much comprehensible information it contains. This criterion is sufficient if the image reader is human.

The other criterion is the mean square error (MSE). MSE is defined mathematically as

$$MSE = \frac{1}{N} \sum_{i,j} [y(i,j) - x(i,j)]^2$$

where x(i,j) = original pixel,

y(i,j) = restored pixel,

N = total number of pixels.

Mostly, this criterion does not indicate how much comprehensible information is restored. This is mainly because the error introduced during restoration is not properly distributed across the restored image. However, due to lack of knowledge of the details of human vision, a mathematical formulation that represents the vision process

is out of reach for the present. Thus, this criterion is acceptable for quantitative comparisons among different algorithms.

For noisy blurred images, the MSE criterion is replaced by the following improvement in dB criterion [2] which is defined as

$$\text{dB improvement} = 10 \log_{10} \frac{\sum_{i,j} [g(i,j) - x(i,j)]^2}{\sum_{i,j} [y(i,j) - x(i,j)]^2}$$

where g(i,j) is the blurred image pixel (i,j).

QUANTITATIVE RESULTS OF SIMULATIONS

The quantitative perfomance of the simulations in a noise-free environment, that are described in the last chapter, are shown in Tables 1 and 2.

Table 1. The quantitative performance in case of linear motion deblurring using differrent algorithms.

ALGORITHMS	MEAN SQUARE ER(MSE)	TIME CONSUMED
Pseudo-inverse	62.63	4 mins
Neural Network	2.22	9 mins
Regularized Iterative	0.71	15 mins
Mine	0.0	7 mins

Table 2. The quantitative performance in case of defocused lens deblurring using differrent algorithms.

ALGPRITHMS	MEAN SQUARE ER(MSE)	TIME CONSUMED
Pseudo-inverse	108008	9 mins
Neural Network	4.89	1 hr 35 mins
Regularized Iterative	1.99	1 hr 15 mins
Mine	0.0	32 mins

Table 3 shows the MSE and the time consumption after each iteration when my algorithm is used.

Table 3. The performance of my algorithm after each iteration.

ITERATION NUMBER	MEAN SQUARE ER(MSE)	TIME CONSUMED
2	3481.078	11 mins
3	5.23.396	14 mins
4	80.081	19 mins
5	12.397	21 mins
6	1.891	26 mins
7	0.437	29 mins
8	0.000	32 mins

COMPARISONS

Inspecting the restored images in Figs. 5 and 6, it is seen that the pseudo-inverse filter fails totally in case of defocused lens blur while it succeeds to some extent in the linear motion case. In my opinion, the reason for the failure is the severity of the defocused lens blur.

In the case of the neural network algorithm, the restoration is better than the pseudo-inverse filter. The restored image is deblurred. The ringing artifacts are not totally suppressed although the suppression is better in the linear motion case.

For the regularized iterative algorithm, ringing is totally suppressed at the expense of restored image sharpness. A close look at Fig.5c reveals some fuzziness in the restored image.

On the other hand, inspecting Figs. 5d and 6d, the restored images using my algorithm, it is found that there are no apparent differences in comparison with the original image "ahmad".

Moreover, the quantitative criterion (MSE) from Tables 1 and 2 shows that my algorithm is indeed not different than the original image in any sense. This criterion also shows that the other algorithms introduce some errors as seen from the tables.

MY ALGORITHM

The criteria used above show that my algorithm restores exactly the original image from the blurred image without any error or approximations. Besides that, it is observed from Tables 1 and 2

that my algorithm consumes less time than any of the other algorithms except for the pseudo-inverse filter. My algorithm consumes about 42% of the time consumed by the regularized iterative method in case of the defocused lens and about 47% in case of linear motion. Compared with the neural network approach, my algorithm consumes about 33% of the time consumed by the former in case of the defocused lens and 78% in case of linear motion. Note that the recorded times in the tables are the shortest times observed on two machines Utek 4319 and Utek 4315.

Moreover, in cases where time is critical, my algorithm could be used satisfactorily for only 6 iterations as Table 3 and Fig.7e suggest. While the table shows some error after 6 iterations, Fig.7e shows no differences. This is due to the tolerance feature of the human eye where it tolerates some error, especially if the error is well distributed. To study the speed of the convergence of my algorithm, the MSE is plotted versus the number of iterations taken from Table 3. The plot, shown in Fig.9, shows the exponential decay of the error and the steepness of the decay after early iterations.

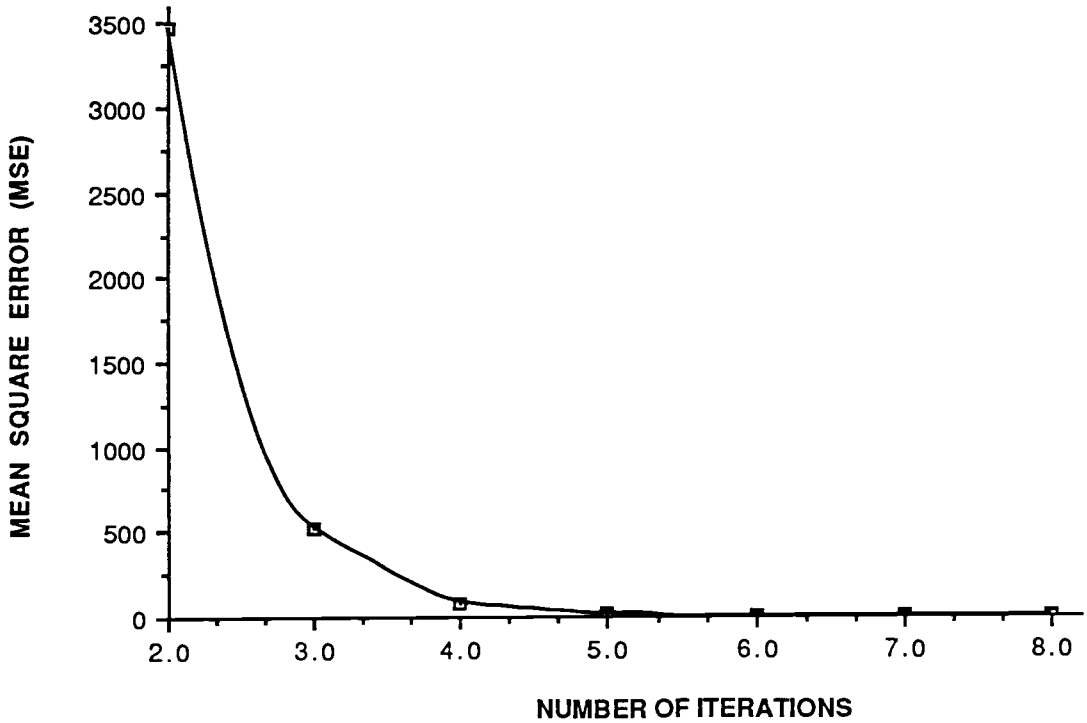


Fig. 9 The convergence of my algorithm.

DISADVANTAGES OF OTHER ALGORITHMS

Besides the relatively long computation time and the approximation of the solution, the neural network and the regularized iterative algorithms exhibit some other disadvantages.

For the neural network approach, the disadvantages are:

1. The algorithm is parameterized. The parameters λ and T are image dependent, and are determined by trial and error.
2. The widths of the sub-border and border regions are image dependent.
3. The ringing artifacts cannot be completely eliminated.

In case of the regularized iterative algorithm, the disadvantages are:

1. The algorithm is also parametrized. α is SNR dependent, β is image dependent and μ is to be chosen.
2. The knowledge of the lowest intensity and the highest intensity in the original image is required.
3. A compromise between deblurring and ringing is inevitable, causing some lack of sharpness in the restored image.

THE EFFECT OF NOISE

For the sake of fairness, I should say that both the neural network and the regularized algorithms are reported to be robust to noise. Using the regularized iterative approach on a blurred image with 50 dB SNR (Fig.8a) my simulation shows that an 8.41 dB improvement in SNR is achieved on the restored image (Fig.8c). Using the neural network approach, I found no improvement in SNR. Fig.8b shows the restored image overwhelmed by noise. I could not find any reason for it where reference [5] reports good smooth restored images.

On the other hand, using my algorithm on the noisy image in Fig.8a does not restore any comprehensible image. In other words, my algorithm cannot tolerate noise. And this is the only disadvantage that I am aware of.

5. CONCLUSION

Since blurring is a convolution process, direct deconvolution is the perfect approach to restore an original image from a blurred image. However, two problems are associated with direct deconvolution. These are singularity in the blur matrix and amplification of noise in noisy environment. These two problems are common and frequent in digital image restoration. This lead researchers to resort to other approaches utilizing estimation theory, optimization theory and others to approximate the solution.

In this thesis, I propose a heuristic algorithm to solve the singularity problem in a noise-free environment. The algorithm utilizes the direct deconvolution approach in association with an iterative scheme to converge to the exact solution. The iterative scheme, with imposed constraints, iterates between the space domain and the frequency domain.

The kind of blurs considered in this thesis are linear motion and defocused lens. Both are assumed to be space-invariant point-spread function (SIPSF) blurs.

The performance of my proposed algorithm is compared against the performance of the pseudo-inverse filter, a neural network approach and a regularized iterative approach. Computer simulations using these algorithms were performed in a noise-free environment. The simulations show that my proposed algorithm is superior to the other algorithms in terms of error and time consumption. It turned out that my algorithm restores images

exactly with no errors and consumes much less time than the non-deconvolution approaches in noise-free environments.

While exact restoration is not necessary for the human eye since it tolerates errors to some extent, it is important for machine vision and wherever details are crucial.

Unfortunately, it is found that my algorithm does not tolerate noise while the other non-deconvolution algorithms are robust to noise. This is the only disadvantage that I am aware of.

Although direct deconvolution techniques are not noise tolerant, I think in the case of singular blur restorations there is a potential to tolerate noise. The potential lies in the singular points in the frequency domain which exhibit some information about the added noise. Investigating such potential and modifying my algorithm to reflect such knowledge will be left for a future research.

BIBLIOGRAPHY

[1]. H. C. Andrews, B. R. Hunt, *Digital Image Restoration*, Prentice-Hall, 1977 .

[2]. J. Biemond, " Stochastic Linear Image Restoration," in *Advances in Computer Vision and Image Processing*, Thomas Huang, Ed., Vol. 2, 1986, Ch. 5 .

[3]. B. R. Hunt, " The Application of Constrained Least Squares Estimation to Image Restoration by Digital Computers," *IEEE Transaction on Computers*, Vol. 22, No. 9, Sept. 1973.

[4]. R. L. Lagendijk, J. Biemond, D. E. Boekee, " Regularized Iterative Image restoration with Ringing Reduction," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol.36, No. 12, pp. 1874-1887, Dec. 1988.

[5]. T. Zhou, R. Chellappa, A. Vaid, B. K. Jenkins, "Image Restoration Using a Neural Network," *IEEE Trans Acoust., Speech, Signal Processing*, Vol.36, No. 7, pp. 1141-1150, July 1988.

[6]. H. V. Sorensen, D. L. Jones, M. T. Heideman, C. S. Burrus, " Real-Valued Fast Fourier Transform Algorithms," *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. 35, No. 6, pp. 849-863, June 1987.